On Iterative Regularization and Its Application

Michael R. Charest Jr.[†] and Peyman Milanfar[§]

Index Terms:

Iterative, Regularization, denoising, film grain, texture transfer, feedback, residual

Abstract

Many existing techniques for image restoration can be expressed in terms of minimizing a particular cost function. Iterative regularization methods are a novel variation on this theme where the cost function is not fixed, but rather refined iteratively at each step. This provides an unprecedented degree of control over the tradeoff between the bias and variance of the image estimate, which can result in improved overall estimation error. This useful property, along with the provable convergence properties of the sequence of estimates produced by these iterative regularization methods lend themselves to a variety of useful applications. In this paper, we introduce a general set of iterative regularization methods, discuss some of their properties and applications, and include examples to illustrate them.

I. Introduction

It is useful to consider perhaps the most typical of image restoration problems[1] first; namely denoising. Typically, one models a measured image as a "true image" plus some error (random noise, film granularity, etc.) We can write this as:

$$\mathbf{y} = \mathbf{x} + \mathbf{v} \tag{1}$$

where $\mathbf{x}$ is the true image that we wish to recover, and $\mathbf{v}$ is the zero-mean noise. The problem of recovering $\mathbf{x}$ from the data $\mathbf{y}$ in this simplified model is the canonical image reconstruction problem of " denoising."

[†] Electrical Engineering Dept., Univ. of California, Santa Cruz, 95064 (email: charest@soe.ucsc.edu)

[§] Electrical Engineering Dept., Univ. of California, Santa Cruz, 95064 (email: milanfar@ee.ucsc.edu)

This work was supported in part by AFOSR grant F49620-03-1-0387

[1]Note that for ease of notation we carry out all of our analysis with vectors representing 1-D signals, though the treatment is valid in multiple dimensions.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JUN 2007		**00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
On Iterative Regularization and Its Application	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California,Electrical Engineering Department,Santa Cruz,CA,95064**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
http://www.soe.ucsc.edu/~milanfar/IterativeRegRevisedJun07.pdf

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**15**	

Denoising Technique	$J(\mathbf{x})$
Tikhonov [1]	$\frac{\lambda}{2}\|\mathbf{x}\|^2$
Total Variation [2], [3]	$\lambda\|\|\nabla\mathbf{x}\|\|_1$
Bilateral Filter [4], [5], [6]	$\frac{\lambda}{2}\sum_{n=-N}^{N}\left[\mathbf{x}-\mathbf{S}^n\mathbf{x}\right]^T\mathbf{W}_{\mathbf{y},n}\left[\mathbf{x}-\mathbf{S}^n\mathbf{x}\right]$

TABLE I

VARIOUS DENOISING TECHNIQUES AND THEIR ASSOCIATED REGULARIZATION TERMS.

A very extensive body of work exists with many different techniques for addressing this problem. We will focus on what are known as " regularization methods," and will specifically present a fundamental extension of these methods to what we call "iterative regularization" methods. Classically, regularization is a very general technique for estimating $\mathbf{x}$ from the data $\mathbf{y}$ by minimizing a cost function of the form:

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} C(\mathbf{x},\mathbf{y}) = \arg\min_{\mathbf{x}} \left\{H(\mathbf{x},\mathbf{y}) + J(\mathbf{x})\right\}. \tag{2}$$

For denoising problems the functional $H(\mathbf{x},\mathbf{y}) = \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|^2$ is typically used, forcing the Euclidean distance between the measured data and the true signal to be minimized. The second term in the above cost function, $J(\mathbf{x})$, is the (generally convex) regularization functional. The regularization functional has the dual purpose of introducing prior information into the solution, encouraging it to have certain desirable properties; and to numerically stabilize the problem, avoiding useless solutions[2]. Generally, the exact form of $J(\mathbf{x})$ varies depending on the particular regularization method, but it usually enforces some additional soft constraint on the estimate. In the Bayesian interpretation of (1), $\mathbf{x}$ is assumed to be a random variable with some prior distribution. In this case, the regularization term, $J(\mathbf{x})$, can be viewed as the log likelihood or *a priori* information about the distribution of $\mathbf{x}$.

Some examples of $J(\mathbf{x})$ for a few different regularization cost functions are given in Table I. The parameter λ in $J(\mathbf{x})$ is known as the regularization parameter and acts as a weight on the prior information. Therefore, the amount of weight given to the prior can be varied by changing λ.

Tikhonov regularization is a classical method ([1]) where the functional ($J(\mathbf{x})$) forces the energy of the estimate to be minimized. Total Variation (TV) regularization is a more recent image denoising method developed originally by Rudin, Osher, and Fatemei [7]. The TV regularization functional forces the L_1-norm of the image gradient to be minimized. This has the effect of making the resulting image appear to

[2]Without the regularization terms, minimizing $H(\mathbf{x},\mathbf{y}) = \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|^2$ for instance, would yield the trivial solution of $\widehat{\mathbf{x}} = \mathbf{y}$!

be piecewise constant. The Bilateral Filter was first proposed as a spatially adaptive filter for denoising by Tomasi and Manduchi in [8]. It was later connected to regularization by Elad in [4]. In the regularization term corresponding to the Bilateral Filter in Table I, $\mathbf{S}^n$ is a matrix shift operator and $\mathbf{W}_{\mathbf{y},n}$ is a matrix of weights where these are a function of both the radiometric (or gray-value) and spatial distances between pixels in a local neighborhood ([4],[5]).

Figure 1 is an example of an estimate of a true image produced by using the Bilateral Filter. By looking at the estimate residual $(\mathbf{y} - \widehat{\mathbf{x}})$ we notice that we have removed some of the high frequency content of the image along with the noise. Similar results can be observed for other regularization methods as well as other general techniques of denoising, as it is never possible to recover $\mathbf{x}$ exactly.

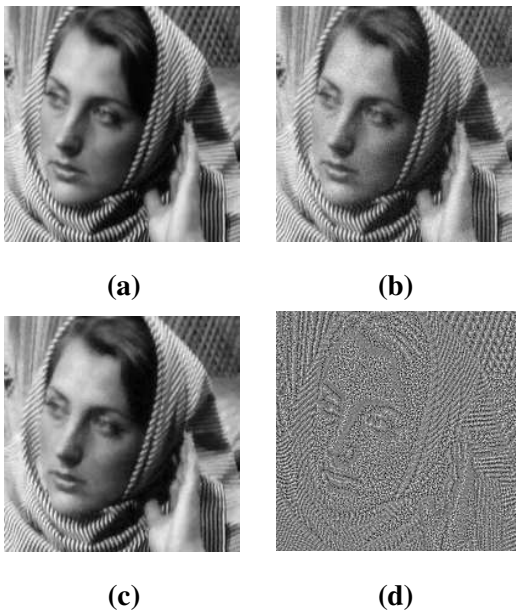


Fig. 1. **(a)** Detail of the original ' Barbara' image **(b)** ' Barbara' with added white Gaussian noise of variance 29.5 (MSE= 29.50) **(c)** The result of minimizing the Bilateral cost function for the noisy image (b) (MSE= 19.30) **(d)** The residual (b)-(c)

There is indeed a natural, and well-known, trade-off between the amount of noise removed, and image detail retained in the regularized estimate. More regularization will remove more noise but will also lose more image detail, and vice versa. Iterative regularization methods presented in this paper exploit this tradeoff to obtain an improved estimate.

II. Iterative Regularization Methods

Iterative regularization methods seek to improve the traditional regularized image estimate by iteratively updating the cost function of choice. Mathematically, this concept can be expressed as follows:

$$\widehat{\mathbf{x}}_{k+1} = \arg\min_{\mathbf{x}} C_k(\mathbf{x}, \mathbf{y}), \tag{3}$$

where by definition, $\widehat{\mathbf{x}}_1$ is the minimizer of (2). The estimate at each iteration can be thought of as the minimizer of a *revised* cost function $C_k(\mathbf{x}, \mathbf{y})$, which is part of a sequence of cost functions updated according to some rule, as we will elucidate in detail below. To summarize, however, the sequence of cost functions is generally constructed using some (possibly nonlinear) combination of the original data $\mathbf{y}$, and the previous estimate $\widehat{\mathbf{x}}_k$. It is important to note at this point that iterative regularization involves two levels of iteration. In particular, at each stage (for each fixed k) a distinct cost function ($C_k(\mathbf{x}, \mathbf{y})$) is minimized, using some iterative method such as steepest descent. In what follows, unless explicitly stated otherwise, these are *not* the iterations to which we refer. Instead, " iteration" below refers to the process of updating the estimate $\widehat{\mathbf{x}}_k$ indexed by k.

The true objective of the iterative regularization approach is that it will yield (for some appropriate stopping index k^* – more on this later), an estimate $\widehat{\mathbf{x}}_{k^*}$ which will have the smallest mean-squared error as compared to all the other members of the sequence $\{\widehat{\mathbf{x}}_k\}$. The main benefit of such an approach is the level of control given to the user by gradually (and iteratively) adding the lost detail back to the initial regularized estimate, but there are a few other incidental benefits as well. Normally the user must select some operating parameters (such as λ) for a particular cost function. The resulting methods that depend strongly on this careful choice of regularization parameter implicitly assume some clairvoyance on the part of the user to produce good results. The iterative approach, on the other hand, allows the initial parameter selection to be essentially arbitrary (that is, one that may yield a poor regularized estimate either visually or in some metric such as mean-squared error) and still produce a satisfactory final estimate after several updates of the cost function, and its subsequent numerical minimization. While this approach certainly results in increased computational complexity, one must keep in mind that automatic methods for selecting parameters such as λ (e.g. cross-validation methods [9]) are also computationally costly, and rather less generally useful.

As we will illustrate later, the proposed iterative methods asymptotically converge back (in the L_2 norm) to the initial data when the initial cost function is $C_1(\mathbf{x}, \mathbf{y}) = \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 + J(\mathbf{x})$, and $J(\mathbf{x})$ is assumed to be a non-negative convex function[3]. This means that by iterating we can gradually "undo"

[3]For proof of convergence of these methods, we refer the interested reader to [10] chapter 4, sections 1-4.

the highly non-linear effect of the initial regularization. This property allows several practical image processing applications of these methods to be realized.

Perhaps one of the earliest proponents of the type of iterative approaches we promote here was the influential statistician John Tukey (1915-2000). Tukey's idea of " twicing", which he published in his 1977 landmark book *Exploratory Data Analysis* ([11]), did not specifically refer to carrying out more than one iteration, though he should be credited with the very original idea. Mathematically, this general "twicing" method can be formulated as:

$$\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_i\right) + J(\mathbf{x}) \right\}, \quad \text{where} \quad \widehat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \mathbf{y}\right) + J(\mathbf{x}) \right\}.$$

As should be apparent from the above, at each iteration, the residual (difference between the estimate and the data) is used to find a " correction" term which improves the previous estimate. For instance, after the first iteration this is most clear: $\widehat{\mathbf{x}}_2 = \widehat{\mathbf{x}}_1 + \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1\right) + J(\mathbf{x}) \right\}$.

More recently, some instances of iterative regularization have been studied in the literature in the past decade (for instance in applications to tomographic reconstruction [12]), but a systematic study of such methods was not undertaken until very recently. In particular, in an effort to patch the undesirable "stair-casing" effect observed in the popular Digital Total Variation (TV) methods [3] (though it should be said that this behavior was observed earlier as well), Osher et al. [2] recently proposed an iterative regularization technique which can be formulated as:

$$\widehat{\mathbf{x}}_{k+1} = \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \mathbf{y} + \sum_{i=1}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i)\right) + J(\mathbf{x}) \right\}.$$

In this framework, in contrast to Tukey's method, the *sum* of the residuals has been added back to the noisy image and the result is processed again. The intuition here is that if, at each iteration, the residual contains more signal than noise, the estimate, $\widehat{\mathbf{x}}_k$, will improve.

In what follows, we present an additional method of iterative regularization, which has interesting properties, specific applications, and proves useful in illuminating the framework in general.

To simplify the presentation, let us define the operator $\mathcal{B}(\cdot)$ to denote the net effect of minimizing the cost function defined in (2); that is: $\mathcal{B}(\mathbf{y}) \equiv \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \mathbf{y}\right) + J\left(\mathbf{x}\right) \right\}$). With this notation in place, we can write the above described methods, and the proposed iterative regularization method in shorthand

as follows:

$$\text{1) Osher et al. [2]:} \quad \widehat{\mathbf{x}}_{k+1} = \mathcal{B}\left(\mathbf{y} + \sum_{i=1}^{k}(\mathbf{y} - \widehat{\mathbf{x}}_i)\right), \tag{4}$$

$$\text{2) Tukey [11]:} \quad \widehat{\mathbf{x}}_{k+1} = \mathcal{B}(\mathbf{y}) + \sum_{i=1}^{k}\mathcal{B}(\mathbf{y} - \widehat{\mathbf{x}}_i), \tag{5}$$

$$\text{3) Proposed:} \quad \widehat{\mathbf{x}}_{k+1} = \mathcal{B}(\mathbf{y}) + \sum_{i=1}^{k}(\mathcal{B}(\mathbf{y}) - \mathcal{B}(\widehat{\mathbf{x}}_i)) \tag{6}$$

$$= (k+1)\mathcal{B}(\mathbf{y}) - \sum_{i=1}^{k}\mathcal{B}(\widehat{\mathbf{x}}_i) = (k+1)\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k}\mathcal{B}(\widehat{\mathbf{x}}_i).$$

As was proved in [10], the above methods share very similar convergence properties, even though they do not generally produce the same minimum MSE results [4].

Summarizing what we have presented so far: (a) we have attempted to make a case for the use of iterative regularization methods in general, where the most notable application of these in denoising has already been demonstrated in [2], [13], and [14]; (b) we related the various forms of iterative regularization and elucidated some connections among them; and (c) we proposed a new approach to iterative regularization – namely 3) above, which was first introduced in [14], [10]. In what follows, we will dissect this proposed method, describe its special properties, and make use of it for some applications.

Proposed Method: Unsharp Residual Iteration (URI) The remainder of this paper is concerned with further analysis of this method and its applications. To establish a connection to some familiar linear filtering techniques, we recall the well-known process of (linear) unsharp masking whereby edges are enhanced by subtracting a blurred version of an image from the image itself [15]. The iterative regularization method that we focus on bears a resemblance to unsharp masking in terms of its structure.

To further elucidate this connection, and to seek a more intuitive presentation of the method, we note that (6) implies:

$$\widehat{\mathbf{x}}_k = k\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k-1}\mathcal{B}(\widehat{\mathbf{x}}_i),$$

[4]We briefly comment that the above methods are related by a linear distribution of $\mathcal{B}(\cdot)$. But since $\mathcal{B}(\cdot)$ is not a linear operator, the methods are therefore quite distinct. Many different variations of these methods can be derived following this framework, as outlined in [10]

which means that we can write

$$\widehat{\mathbf{x}}_{k+1} = (k+1)\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k} \mathcal{B}(\widehat{\mathbf{x}}_i) = k\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k-1} \mathcal{B}(\widehat{\mathbf{x}}_i) + (\widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_k)) \quad (7)$$

$$= \widehat{\mathbf{x}}_k + (\widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_k)) = \widehat{\mathbf{x}}_1 + (\widehat{\mathbf{x}}_k - \mathcal{B}(\widehat{\mathbf{x}}_k)) \,. \quad (8)$$

In (8) above, the " residual" term which updates the original estimate $\widehat{\mathbf{x}}_1$ to yield $\widehat{\mathbf{x}}_{k+1}$ is the difference between the last estimate $\widehat{\mathbf{x}}_k$ and a nonlinearly "filtered" version of it, namely $\mathcal{B}(\widehat{\mathbf{x}}_k)$. Hence, we propose to call the method "Unsharp Residual Iteration" (URI). We illustrate this method in Figure 2 using a block diagram, where we define $\widetilde{\mathbf{x}}_k \equiv \mathcal{B}(\widehat{\mathbf{x}}_k)$.

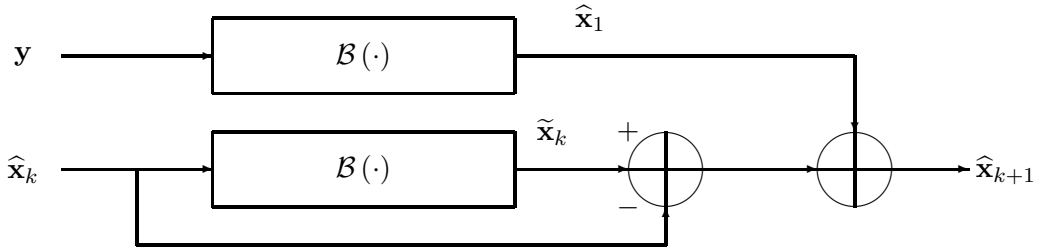


Fig. 2. URI Block Diagram

As an illustrative example, we show the effect of the URI procedure on a portion of the noisy Barbara image in Figure 3. In this example, where the regularization function $J(\mathbf{x})$ was selected as the digital TV term (but can be any convex functional) introduced in Table I, the initial estimate $\widehat{\mathbf{x}}_1$, along with the next two iterates, and the corresponding residuals, are shown. For this example, the regularization parameters were selected to produce the lowest overall mean-squared error estimate. The smallest MSE estimate occurs at the second iteration. This is typical in that the smallest MSE estimate is usually arrived at in fewer than five iterations. In practical use for applications such as denoising [14] when the underlying image is unknown, a question worthy of further research is the matter of finding a stopping criterion which will automatically end the iterations when the optimal MSE is reached. Such a criterion may for instance be constructed based on the statistics of the residual terms. Here, similar to [2], we use an estimate of variance of the residual images to propose an empirical stopping criterion.

With URI (as well as the other iterative regularization methods) some lost detail is added back to the image estimate at each iteration, and simultaneously, some noise is also added to the image estimate at each iteration. The general trend of the estimate $\widehat{\mathbf{x}}_k$ (as proved in [10] and illustrated in Figure 5) is to monotonically approach the true image $\mathbf{x}$ first (with corresponding monotonic decrease in MSE); then, when the noise content of the estimate $\widetilde{\mathbf{x}}_k$ drops below a critical level (in comparison to the level of

lost detail), the estimate $\widehat{\mathbf{x}}_k$ begins to asymptotically approach the noisy image $\mathbf{y}$ (with corresponding increase in the MSE).

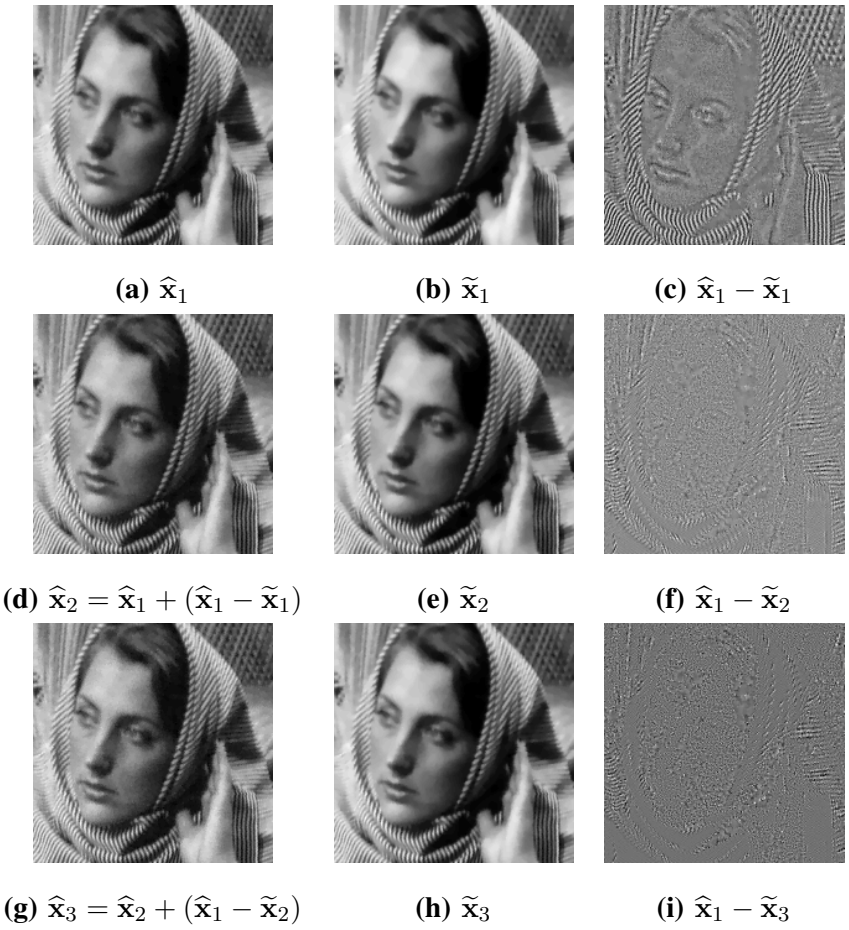
We can look at $\|\mathbf{y}-\widehat{\mathbf{x}}_k\|^2$ as a way to gauge how close the image estimate is to the noisy measured image $\mathbf{y}$. With an arbitrary initial guess, the value of this metric will generally start out high and decrease as the image estimate begins to approach the underlying image $\mathbf{x}$. Assuming that $\mathbf{y}$ is corrupted by Gaussian white noise, as $\widehat{\mathbf{x}}_k$ approaches $\mathbf{x}$, $\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 \to \delta^2$, where δ^2 is the variance of the corrupting noise in $\mathbf{y}$. Thus, following [2], we propose $\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 \leq \delta^2$ as an empirical stopping criterion for URI. While the noise level δ^2 may not be directly known to the user, an estimate of it can generally be computed.

All the examples illustrated in the paper have used the same stopping criterion where applicable. To independently illustrate the use of this stopping criterion, we present another example. White Gaussian noise with a variance of 50 was added to the image in Figure 4a to produce Figure 4b. Next, 10 iterations of URI were applied to Figure 4b using the bilateral filter regularization with the following parameters: $N = 2$, $\sigma_r = 110$, and $\sigma_d = 1.5$. For completeness, the best estimate, $\widehat{\mathbf{x}}_4$ is shown in Figure 4c. In Figure 4d we see that the best (lowest MSE) estimate occurs at iteration 4. The metric used to determine the stopping criterion is plotted against iteration number in Figure 4e. The intersection of this curve with the horizontal line representing the value of δ^2 indicates that the iterative regularization process should be stopped at 4 iterations. Thus we observe that the the stopping criterion has accurately indicated when to stop iterating.

To measure the behavior of any algorithm in recovering the true signal $\mathbf{x}$ from the data $\mathbf{y}$, the mean-squared error (MSE) is a natural choice. The MSE is defined as $\mathbf{mse}(\widehat{\mathbf{x}}) = E\left[(\widehat{\mathbf{x}} - \mathbf{x})^2\right]$, where $\widehat{\mathbf{x}}$ is the estimate and $\mathbf{x}$ is the underlying signal. As is well-known, we can rewrite the MSE as

$$\begin{aligned}
\mathbf{mse}(\widehat{\mathbf{x}}) &= E\left([(\widehat{\mathbf{x}} - E(\widehat{\mathbf{x}})) + (E(\widehat{\mathbf{x}}) - \mathbf{x})]^2\right) \\
&= E\left[(\widehat{\mathbf{x}} - E(\widehat{\mathbf{x}}))^2\right] + 2\,E\left[(\widehat{\mathbf{x}} - E(\widehat{\mathbf{x}}))\,(E(\widehat{\mathbf{x}}) - \mathbf{x})\right] + E\left[(E(\widehat{\mathbf{x}}) - \mathbf{x})^2\right] \\
&= \mathbf{var}(\widehat{\mathbf{x}}) + 0 + (E(\widehat{\mathbf{x}}) - \mathbf{x})^2 = \mathbf{var}(\widehat{\mathbf{x}}) + \mathbf{bias}^2(\mathbf{x}),
\end{aligned}$$

which illustrates the useful fact that MSE is the sum of the estimate variance and squared-bias [16]. We use the next experiment to show the bias-variance properties of URI experimentally using Monte-Carlo simulations. The other iterative regularization methods mentioned above have quite similar bias-variance tradeoffs and these have been catalogued extensively in [14], [10]. We mention again here that in all cases, the bias tends toward zero with increasing number of iterations, while the variance tends to converge to the variance of the corrupting noise. However, as alluded to earlier, these methods do not produce the same minimum MSE estimates, and their relative performance is highly dependant on the image content,



(a) $\widehat{\mathbf{x}}_1$ **(b)** $\widetilde{\mathbf{x}}_1$ **(c)** $\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_1$

(d) $\widehat{\mathbf{x}}_2 = \widehat{\mathbf{x}}_1 + (\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_1)$ **(e)** $\widetilde{\mathbf{x}}_2$ **(f)** $\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_2$

(g) $\widehat{\mathbf{x}}_3 = \widehat{\mathbf{x}}_2 + (\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_2)$ **(h)** $\widetilde{\mathbf{x}}_3$ **(i)** $\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_3$

Fig. 3. Noisy data is the same as that in Figure 1. **(a)** The first estimate produced by URI with Total Variation regularization (MSE=26.76). **(b)** The estimate produced from (a). **(c)** The residual between (a) and (b). **(d)** The second estimate produced by URI (MSE=18.09) from summing (a) and (c). **(e)** The estimate produced from (d). **(f)** The residual between (a) and (e). **(g)** The third estimate produced by URI method (MSE=20.22) from summing (a), (c), and (f). **(h)** The estimate produced from (g). **(i)** The residual between (a) and (h).

and the noise variance, and the regularization functional.

To carry out these Monte-Carlo simulations, we add a realization of random (Gaussian) noise with a variance of 29.5 to the 'Barbara'image. A series of estimates ($\widehat{\mathbf{x}}_1$ through $\widehat{\mathbf{x}}_{10}$) are computed from this noisy image using the proposed URI method (8). This process is repeated for a total of 50 different noise realizations. Next, the average variance, bias, and MSE are calculated from these realizations at each iteration ($\widehat{\mathbf{x}}_1$ through $\widehat{\mathbf{x}}_{10}$).

In Figure 5 we have plotted the average MSE, variance, and squared-bias of URI as a function of iteration number. Two different regularization functionals (Total Variation and Bilateral Filter) were used

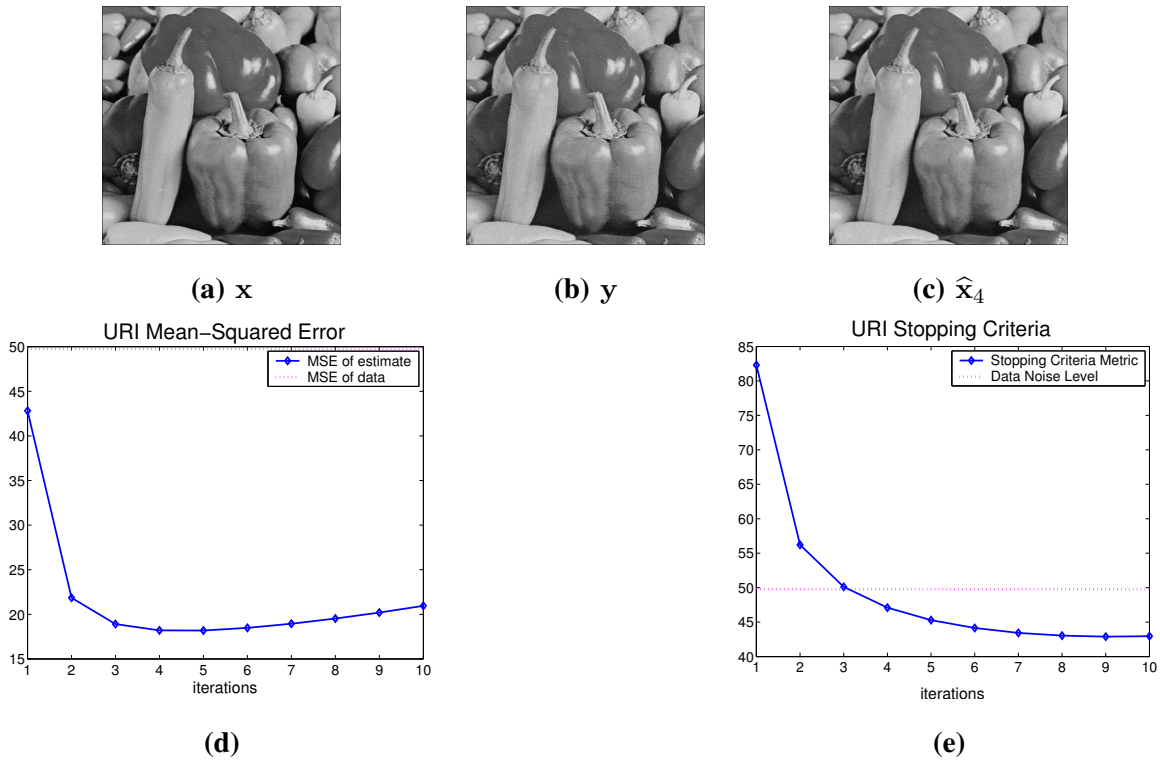
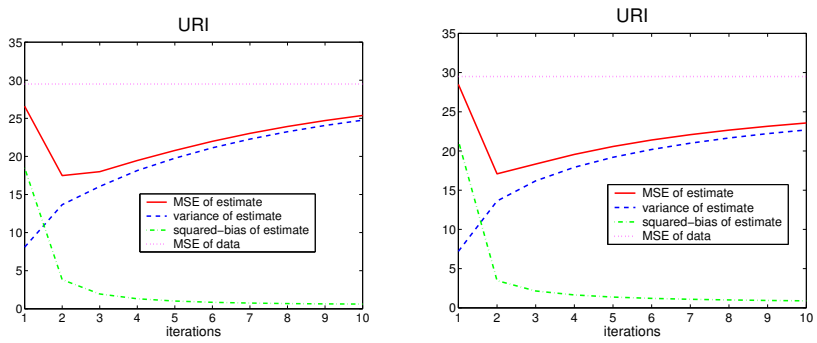


Fig. 4. **(a)** The original image. **(b)** The image corrupted with noise of variance $\delta^2 = 50$. **(c)** The best estimate for the given parameters. **(d)** Plot of the mean-squared error of $\widehat{\mathbf{x}}_k$ vs. iteration. **(e)** Plot of the stopping criterion vs. iteration.

for comparison. The operating parameters for each of the regularization functionals were selected to yield the overall lowest MSE. For the Bilateral [8] regularization we used kernel size $N = 2$ (5×5), spatial blur parameter $\sigma_d = 1.1$, and radiometric blur parameter $\sigma_r = 23$. For the Total Variation Regularization [3] we used regularization parameter $\lambda = .32$ and 50 steepest descent iterations[5]. For details on the implementation of either of these regularization methods, we refer the reader to [8] and [3], respectively.

Interestingly, we also observed that the best overall MSE estimate is arrived at by initially applying a relatively large amount of regularization such that the first estimate has a low variance (due to the removal of noise) but a high bias (due to the loss of high frequency image details). The bias then decreases as we iterate and more of the " lost detail" is returned to the estimate. Some noise is also returned along with the detail, causing the variance to increase. The MSE, being the sum of these two statistics is optimal at the point in the iterative process where we get the best tradeoff between restored texture and suppressed

[5]Note that the steepest descent iterations are performed for each optimization of a cost function (e.g. 50 steepest descent iterations are done for each URI iteration)



(a) Bilateral **(b)** Total Variation

Fig. 5. Average MSE, variance, and squared-bias of the estimates $\widehat{\mathbf{x}}_k$ of the noisy versions of the 'Barbara' image, using URI with (a) Bilateral and (b) Total Variation regularization functionals.

noise.

As a comparison between URI and the methods of Osher et al. [2] and Tukey [11] we include an experiment (using the Bilateral Filter as the regularization functional) in Figure 8. The image tested was 'Barbara' with white Gaussian noise of variance 29.5 added just as in the above experiments. The Bilateral Filter kernel size for all of the methods was fixed to $N = 2$ and the radiometric and spatial blur parameters were selected to produce the best overall result respectively.

III. Application to Texture and Grain Manipulation and Transfer

The ability to efficiently transfer the texture from one image to another finds uses in a variety of areas [17]. For artistic effect, it may be desirable to add grainy texture to images captured on digital media. Composite images may combine elements from different photographs into one; or sometimes a computer generated image is combined with a real one. It is often desirable that the perceived texture across the composite image be consistent. Also, high frequency textures such as hair or sand can be difficult or labor intensive to implement in computer generated images. In some instances it would be advantageous to simply extract the texture from a real image and transfer it to another (possibly computer generated) image. Aside from published academic work (see [17] for an overview,) commercial products such as 'Grain Surgery' (*http://www.visinf.com/gs/ps/*) have also been developed to accomplish these types of tasks. Our method offers a computationally attractive alternative.

The simplest (but not most effective) way to accomplish the texture transfer is to apply either Total Variation or Bilateral Filter Regularization to the source image $\mathbf{y}$ containing the desired texture, resulting

in an approximately piecewise-constant image $\widehat{\mathbf{x}}_1$. The texture information $(\mathbf{y} - \widehat{\mathbf{x}}_1)$ thus captured can then be added to the textureless image $\mathbf{z}$ with proper normalization. We can express this process as:

$$\widehat{\mathbf{z}} = \mathbf{z} + \mathcal{N}(\mathbf{y} - \widehat{\mathbf{x}}_1), \tag{9}$$

where $\mathcal{N}(\cdot)$ is a normalization of the image gray values such that the texture and target images are in the same proper dynamic range. Instead of this brute-force method, there is a significant advantage to using URI to transfer the texture; namely more control. By transferring texture from image $\mathbf{y}$ to image $\mathbf{z}$ as:

$$\widehat{\mathbf{z}}_k = \mathbf{z} + \mathcal{N}\left(\sum_{i=1}^{k}(\widehat{\mathbf{x}}_i - \widetilde{\mathbf{x}}_i)\right) \tag{10}$$

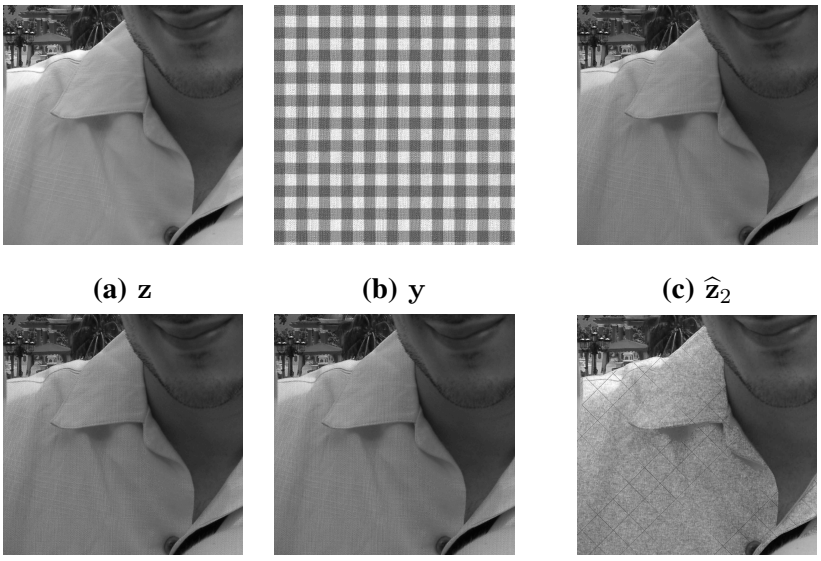
we can control the amount of texture that is transferred, easily optimizing the appearance of the resulting image.

In Figure 6 we show an example of this technique. We use Bilateral Filter regularization with kernel size $N = 2$, spatial parameter $\sigma_d = 1.1$, and radiometric parameter $\sigma_r = 50$ to iteratively add the texture from a texture source image (Figure 6b) to the target image in Figure 6a. A simple intensity threshold was used to define the region of interest receiving texture. Notice how the transferred texture becomes more pronounced as the number of iterations increases. Figure 6f is an example of texture transfer using the a trial version of the commercially available software 'Grain Surgery' mentioned above.

To demonstrate another example and application, we present in Figure 7 an instance where film grain has been removed from a scanned 35-mm photo, yielding a " cartoon" version of the image. Then the grain texture is returned to a high-quality compressed version of this cartoon, by iterative application of URI without direct knowledge of the original image. For both of the above examples, the regularization parameter λ was selected such that the most visually appealing results were obtained.

IV. Conclusions

Iterated regularization presents a very general methodology for image decomposition and reconstruction. Under this framework we have related the independently derived methods of Osher et al. [2] and Tukey [11] and proposed a new method. We illustrated this proposed method as well-suited to applications in texture and grain synthesis and transfer. Generally, each of the iterative regularization methods describes a technique for gradually returning the removed detail to an initially nonlinearly filtered estimate. The true generality of the framework lies in the fact that *any* form of regularization may be used in this context. One can think of any image restoration method as simply a black box, where the presented framework yields several ways to feed back the estimate into this black box and improve the overall reconstruction quality.



(a) $\mathbf{z}$ **(b)** $\mathbf{y}$ **(c)** $\widehat{\mathbf{z}}_2$

(d) $\widehat{\mathbf{z}}_4$ **(e)** $\widehat{\mathbf{z}}_{10}$ **(f)** 'Grain Surgery' result

Fig. 6. **(a)** The original image without any texture added. **(b)** The texture source. **(c)** The result of one URI iteration (with Bilateral Filter Regularization) to transfer the texture of the image in (b) to the image in (a). **(d)** The result of three URI iterations. **(e)** The result of nine URI iterations. **(f)** Transfer of same texture using commercially available software (cross-hatch pattern is added as part of the trial version of the software).

A question that is relevant to all the above methods is " What *overall* cost function do these iterative methods minimize?" That is to say, is the iterative regularization implicitly minimizing some ("hidden"), global, cost function that is not explicitly given? This question remains an interesting open problem.

Finally, though this paper only dealt with the noisy data model (1), it is possible (as briefly described in [14]) to further extend the analysis presented here to the more general case, where the data ($\mathbf{y}$) is modeled as $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{v}$ where $\mathbf{A}$ is a convolution (blur) operator. Some initial results on this idea are reported in [10], [14].

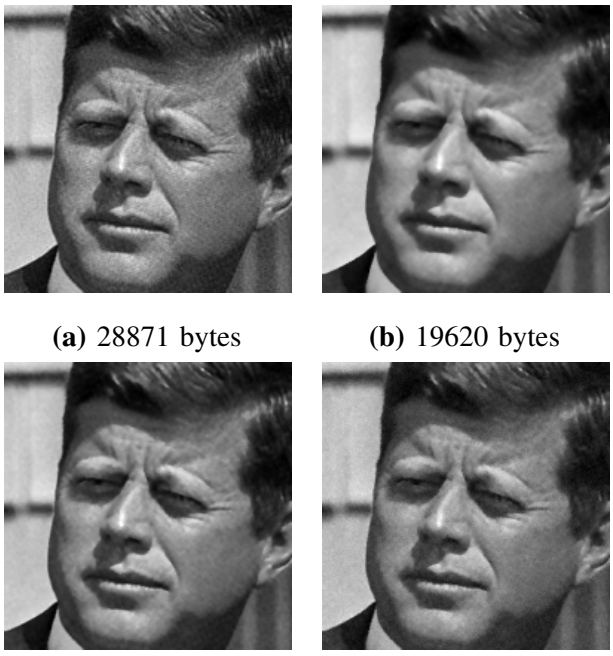
REFERENCES

[1] A. Tikhonov, "On the stability of inverse problems," *Dokl. Akad. Nauk SSSR*, vol. 39, no. 5, pp. 195–198, 1943.

[2] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *SIAM Multiscale Model. and Simu.*, vol. 4, pp. 460–489, 2005.

[3] T. Chan, S. Osher, and J. Shen, "The digital TV filter and nonlinear denoising," *IEEE Trans. on Image Processing*, vol. 10, no. 2, pp. 231–241, 2001.

[4] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Trans. on Image Processing*, vol. 11, no. 10, pp. 1141–1151, 2002.



(a) 28871 bytes **(b)** 19620 bytes

(c) **(d)**

Fig. 7. **(a)** The original grainy image $\mathbf{y}$, compressed using JPEG with a quality factor of 100. **(b)** The Bilateral Filtered image $\widehat{\mathbf{x}}_1 = \mathcal{B}(\mathbf{y})$ compressed using JPEG with a quality factor of 100. **(c)** The result of using one iteration of URI on (b). **(d)** The resulting image $\widehat{\mathbf{y}}$ with reconstructed grain texture.

[5] S. Farsiu, M. Robinson, M. Elad, and P. Milanfar, "Fast and robust mulitframe super resolution," *IEEE Trans. on Image Processing*, vol. 13, no. 10, pp. 1327–1344, 2004.

[6] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Advances and challenges in super-resolution," *International Journal of Imaging Systems and Technology*, vol. 14, no. 2, pp. 47–57, Aug. 2004.

[7] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, pp. 259–268, 1992.

[8] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the IEEE International Conference on Computer Vision*, 1998, pp. 836–846.

[9] N. Nguyen, P. Milanfar, and G. Golub, "Efficient generalized cross-validation with applications to parametric image restoration and resolution enhancement," *IEEE Trans. on Image Proc.*, vol. 10, no. 9, pp. 1299–1308, Sept. 2001.

[10] M. Charest, "A general framework for iterative regularization in image processing," Master's thesis, EE Dept., Univeristy of California, Santa Cruz, 2006, available at http://www.soe.ucsc.edu/~milanfar/CharestThesis.pdf.

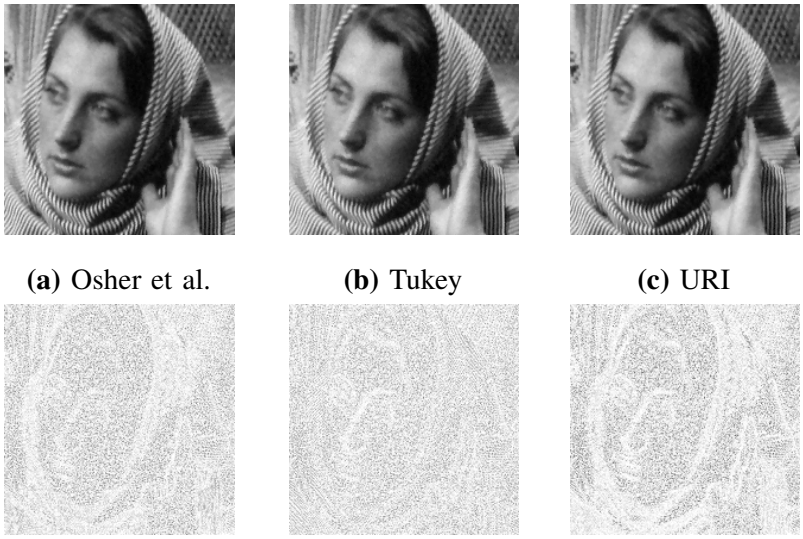
[11] J. Tukey, *Exploratory Data Analysis*. Addison-Wesley, 1977.

[12] P. Milanfar, W. Karl, and A. Willsky, "A moment-based variational approach to tomographic reconstruction," *IEEE Trans. on Image Proc.*, vol. 5, no. 3, pp. 459–470, March 1996.

[13] J. Xu and S. Osher, "Iterative regularization and nonlinear inverse scale space applied to wavelet denoising," *IEEE Trans. on Image Proc.*, vol. 16, no. 2, pp. 534–544, February 2007.

	N	σ_d	σ_r	**iterations**
Osher et al. [2]	2	1.1	35	2
Tukey [11]	2	1.1	14	2
URI	2	1.1	23	2

Bilateral Filter parameters used to generate the estimates below



(a) Osher et al. **(b)** Tukey **(c)** URI

(d) Avg. MSE=18.12 **(e)** Avg. MSE=18.27 **(f)** Avg. MSE=17.48

Fig. 8. Detail of the best MSE estimates of the noisy 'Barbara' using the Bilateral Filter and iterating via: **(a)** The method of Osher et al., **(b)** The method of Tukey, and **(c)** URI. **(d)** The residual of (a) **(e)** The residual of (b) **(f)** The residual of (c).

[14] M. Charest, M. Elad, and P. Milanfar, "A general iterative regularization framework for image denoising," in *Proceedings of the 40th Conference on Information Sciences and Systems,Princeton, NJ*, 2006.

[15] R. Gonzalez and R. Woods, *Digital Image Processing, second edition*. Pearson, 2002.

[16] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, A. Oppenheim, Ed. Prentice Hall PTR, 1993.

[17] M. Ashikhmin, "Fast texture transfer," *IEEE Computer Graphics and Applications*, vol. 23, no. 4, pp. 38–43, July/August 2003.